

## **REMARKS**

Original Claims 1-34 have been examined. All of the claims have been rejected as unpatentable over the prior art of record. Specifically, claims 1-4, 7-17, 20-26, 29, and 32 are rejected as anticipated by U.S. Patent No. 5,819,093 to Davidson (herein "Davidson"), and claims 5, 6, 18, 19, 27, 28, 30, and 31 are rejected as obvious over Davidson. Claims 33 and 34 are rejected as obvious over Davidson in view of U.S. Patent No. 6,199,099 to Gershman (herein "Gershman").

### **Claim Objection—Claim 32**

Claim 32 is objected to for minor informalities. In response, Applicants amended claim 32, as suggested by the Office. Accordingly, Applicants request that the objection to claim 32 be withdrawn.

### **Section 112 Rejection—Claims 1-22 and 32**

Claims 1-22 and 32 are rejected under 35 U.S.C. § 112, first and second paragraphs. In response, Applicants amended claims 1, 15, and 32 to overcome the rejection. In addition to several cosmetic changes to these claims, which do not narrow the scope of the claims, Applicants have also deleted any reference of "independent of a breakpoint" from claims 1 and 15, and claim 32 is amended, as suggested by the Office, to correct the cited minor informalities. Accordingly, Applicants request that the rejection of claims 1-22 and 32 be withdrawn.

### **Section 102 Rejection—Claims 1-4, 7-17, 20-26, 29, and 32**

#### **DAVIDSON DOES NOT DISCLOSE RUNNING THE DEBUGGER IN THE THREAD OF EXECUTION BEING DEBUGGED**

Claims 1-4, 7-17, 20-26, 29, and 32 are rejected under 35 U.S.C. § 102(b) as being anticipated by Davidson. In response, Applicants amended claims 1, 26, and 32 to more clearly recite the feature of the debugger being loaded into the thread of execution of the program being debugged by the debugger, which is not disclosed by Davidson.

Specifically, because a server in Davidson may be servicing many clients, there may be many executions of a particular function occurring in the server (Davidson, Col. 10, lines 60-62). As a result, in order to find the execution that relates to the client being debugged, a thread in the server that was servicing the function call of the client being debugged is identified in Davidson (Davidson, Col. 10, lines 60-62). However, this thread cited by the Office is a thread executed in the server, and not a thread executed in the client being debugged. In contrast, the thread, as recited in the claims, is in the program being debugged, specifically, not a thread executed in the server as required by Davidson.

Moreover, Davidson further fails to disclose a debugger being loaded into the thread of the program being debugged, because in Davidson, the debugger is not being loaded into the identified thread disclosed in Col. 10, lines 60-62. Rather, Column 10, line 52 through Column 12, line 15 of Davidson is simply a detailed description of FIG. 14. Specifically, FIG. 14 of Davidson depicts (1) a local host 520 having a debugger-GUI 502, a dbx engine 504, a helper process 506, a client side wrapper server 510 and a client 508 and (2) a remote host 522 containing a dbx engine 512, a helper process 514, a server side wrapper server 518 and the server 516 (Davidson, Col. 11, lines 33-36). However, none of these components are described as being loaded into the thread of the program being debugged. The only reference of a thread being used in FIG. 14 in Davidson is the wrapper server creating two new threads on the remote host (Davidson, Col. 11, lines 64-67). However, this is not the same as the recited language of the debugger being loaded into the thread of the program being debugged. For all these reasons, the cited reference does not disclose a debugger being loaded into the thread of execution of the program being debugged as recited in the claims 1, 26, and 32. Applicants accordingly request that the rejection of claims 1, 26, and 32 be withdrawn.

Claim 15 recites as follows:

A method for debugging a program on a computer comprising:  
halting a thread of execution of the program being debugged, wherein the thread is associated with context data describing context of the thread;  
obtaining a pointer to an interface of an object of the program from the context data; and

referencing the pointer to make function calls manually to the interface from within the context of the thread being debugged.

Applicants reassert the above arguments relating to Davidson, and submit that since Davidson does not disclose loading a debugger into the thread that is being debugged, Davidson similarly cannot disclose "*referencing the pointer to make function calls manually to the interface from within the context of the thread being debugged*" as recited, because no functions are being called from within the thread being debugged. Specifically, although Davidson discloses obtaining and referencing the pointer from context data to make a function call to an object from within the *process*, claim 15 does not generally recite this feature. Rather, claim 15 specifically recites that the reference is made from *within the context of the thread being debugged*. There is nothing from Davidson that relates the "process" disclosed in Davidson to the thread being debugged recited in the claims. Similarly, because Davidson does not disclose a debugger being loaded into the thread being debugged or making function calls from within the context of the thread being debugged, Davidson also fails to disclose, among other things, a debugger module operating within the context of the thread of the program being debugged as recited in claim 23.

For all these reasons, Applicants submit that Davidson does not disclose (1) loading and running a debugger into a thread of execution of the program being debugged as recited in independent claims 1, 26, and 32, (2) referencing the pointer to make function calls manually to the interface from within the context of the thread being debugged as recited in independent claim 15, and (3) a debugger module operating within the context of the thread of the program being debugged as recited in independent claim 23. Because the remaining claims 2-4, 7-14, 16, 17, 20-22, 24, 25, and 29 are dependent claims of claims 1, 15, 23, and 26, they are patentable for at least the reasons set forth above with regard to their corresponding independent claims. Nevertheless, Applicants reserve the right to present further arguments in the future with regard to the dependent claims in the event that the independent claims are found to be unpatentable. Accordingly, Applicants request that the Section 102 rejection of claims 1-4, 7-17, 20-26, 29, and 32 be withdrawn.

**Section 103 Rejection—Claims 5, 6, 18, 19, 27, 28, 30, and 31**

Claims 5, 6, 18, 19, 27, 28, 30, and 31 are rejected under 35 U.S.C. § 103(a) as obvious over Davidson. Because claims 6 and 19 have been canceled, the Section 103 rejection of claims 6 and 19 is now moot. Furthermore, since the remaining claims 5, 18, 27, 28, 30, and 31 depend from claims 1, 15, and 26, respectively, they are patentable for at least the reasons set forth above with regard to their corresponding independent claims. Specifically, for the reasons stated in the remarks relating to claims 1, 15, and 26, the combination, whether proper or improper, fails to teach the expressly recited claim elements, and therefore cannot as a matter of law make obvious claims 5, 18, 27, 28, 30, and 31.

Moreover, Davidson further fails to disclose or suggest the recited features of the claims, because it would not be obvious to modify the device disclosed in Davidson. Specifically, no motivation or suggestion can be drawn from Davidson, because the proposed modification would render Davidson inappropriate for its intended purpose. And there is no reasonable likelihood of success even if such a modification is made, because Davidson requires that the thread be on the server. In particular, Davidson requires that the debugging be performed on a server in a distributed environment, whereas the COM and DCOM objects, as recited in claims 5 and 18, are run locally in an operating system. Thus, Applicants respectfully disagree that any suggestion or motivation can be drawn from Davidson to use the COM and DCOM objects, when in fact Davidson specifically teaches away from a debugging system that uses locally ran objects, because Davidson requires that the thread be on the server and does not relate to the actual client that is being debugged. Thus, Davidson does not disclose a COM object or a DCOM object as recited in claims 5 and 18. Moreover, any teachings of the COM and DCOM objects in claims 5 and 18 and the web server in claims 27, 28, 30, and 31 are derived from the specification, and such disclosures are not found in Davidson. Thus, the Office improperly engaged in hindsight reconstruction in using the specification of the invention to guide the modification of Davidson. As a result, it cannot be reasonably said that the motivation or suggestion is derived from Davidson to make the modification

asserted by the Office. Accordingly, for all these reasons, Applicants request that the Section 103 rejection to claims 5, 18, 27, 28, 30, and 31 be withdrawn.

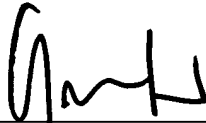
**Section 103 Rejection—Claims 33 and 34**

Claims 33 and 34 are rejected under 35 U.S.C. § 103(a) as obvious over Davidson in view Gershman. In response, Applicants amended independent claim 33, and traverse the rejection. For the reasons stated in the remarks relating to claim 1, the combination, whether proper or improper, fails to teach the expressly recited claim elements, and therefore cannot as a matter of law make obvious claim 33. Specifically, the cited references do not, alone or combined, disclose the features of "loading a debugger into the thread of the ASP module being debugged" and "running the debugger in the thread being debugged to debug the ASP module" as recited in now amended claim 33. Moreover, claim 34 depends from claim 33, and thus it is patentable for at least the reasons set forth above with regard to claim 33. Nevertheless, Applicants reserve the right to present further arguments in the future with regard to dependent claim 34 in the event that claim 33 is found to be unpatentable. Accordingly, Applicants request that the Section 103 rejection of claims 33 and 34 be withdrawn.

**CONCLUSION**

The application is considered in good and proper form for allowance, and the Examiner is respectfully requested to pass this application to issue. If, in the opinion of the Examiner, a telephone conference would expedite the prosecution of the subject application, the Examiner is invited to call the undersigned attorney.

Respectfully submitted,



---

Grace Law, Reg. No. 48,872  
One of the Attorneys for Applicants  
LEYDIG, VOIT & MAYER, LTD.  
Two Prudential Plaza, Suite 4900  
180 North Stetson  
Chicago, Illinois 60601-6780  
(312) 616-5600 (telephone)  
(312) 616-5700 (facsimile)

Date: April 1, 2004